Project Title – Design Document

Team Name

|  |
| --- |
| Members |
|  |
|  |
|  |
|  |
|  |

(This page does not count in the total, and is not marked)

HAND IN DATE: MONDAY 5 NOVEMBER 16:00 via Turnitin

MAX: 15 pages excluding this page, reference list and appendices

Introduction (5%)

A brief description (a few paragraphs) on the purposes of the document and the system to be developed and who it will serve). We will roughly follow the headings of Kruchten’s 4+1 architecture model but we are in ‘start-up’ mode so we don’t have time for doing this exhaustively. In that case the sections below are using the headings from the 4+1 but the content is adapted. You can reorder these sections if you want. Please consider individual background and research to agree what your eventual system will do.

User Stories/Scenarios

In this section, describe the architecture using a small set of scenarios but also review the use of design thinking for this purpose. Please carry out some research into Stanford Design thinking (as mentioned by Peter Edgar in his lecture) and use this to describe the system planned using personas. There are plenty of guides and tutorials on that online. (Leave out the prototype and test steps for this report).

About Design Thinking (10%)

Review here how you employed Design Thinking in this. How were you able to be creative? What particular techniques helped? Give a collective critical view of the approach and how useful it was? Is there any other evidence to support (or oppose) the use of design thinking in in ideation in a software context and in establishing user stories/architectures?

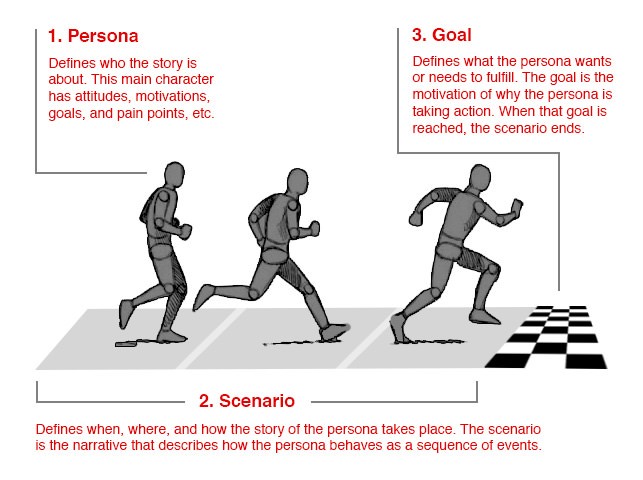


Figure 1: Personas in Design Thinking [1]

Scenarios (30%)

Give detail of the various key scenarios that describe the system. Identify those that will form the MVP and those that are more aspirational (would like to include).

Logical view (15%)

How will the system be partitioned – you can use your own diagram style or UML (package or component diagram). You don’t need an elaborate tool – something like draw.io or visio should be fine.

In this section most important layers/subsystems/packages/classes etc and how these will provide the important functionality.

A class diagrams may be useful here, but not too detailed, just enough to explain the architecture and the main functions. The aim is to get across the functionality that the system provides to end-users.

Physical view (5%)

We are going to be brief in this section, because we will know more about this later in the process. The physical view depicts the system from a system engineer's point of view. It is concerned with the topology of software components on the physical layer as well as the physical connections between these components. This view is also known as the deployment view. Briefly state how the system will be deployed. You could refer to a UML deployment diagram.

Process view (20%)

In the process view we want to concentrate on non-functional requirements ie. quality attributes such as performance, reliability, maintainability, interoperability, scalability, security, concurrency, … etc. . You can use interaction diagrams (sequence diagrams/collaboration diagrams) if you they help to explain but the main thing is to demonstrate the key quality attributes required in the system and to explain how your chosen architecture will assist in meeting them (tactics).

Development View (15%)

How will you implement the system – environment, languages, libraries, architectural styles/patterns etc

What software process will you use and why?

How will the work be split between the team?

How will progress and deliverables be managed and maintained. What protocols will you have for dividing up work – justify your choices?

What tools will you use and why?

How will the work be partitioned – you can refer to the partition of the system in the next section.

References

A list of references to documents (books, papers, web pages etc.) which are referred to in the main body of the text. Use the IEEE citation style as detailed here <https://ieee-dataport.org/sites/default/files/analysis/27/IEEE%20Citation%20Guidelines.pdf>.There is some guidance on referencing at <http://www.qub.ac.uk/cite2write/home.html>.

This is not counted in the page limit and the marks are implicit in other sections. Others’ work should be cited. References should be listed:

1. Interaction Design Foundation, “Personas – a Simple Introduction”, available at <https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them>, accessed 19 October 2018.
2. Add relevant references – you should have a good range about design thinking and also architecture …

Appendices

Anything that breaks up the flow, but keep in mind this is not marked. It may be referred to briefly by the assessor.

Appendix 1

Describe who (or what percentage of) did the work for which sections of the document. Please discuss and agree this section at the very start and at the very end.

Table 1: Arbitrary Table – above the table

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

Appendix 2 – Marking Guidance - Delete from your document

We will employ conceptual marking in this report. This is to give you information on how to get a first class mark and also to avoid a fail. The table below shows you how these two categories will be judged.

|  |  |  |  |
| --- | --- | --- | --- |
| First |  | Fail | |
| **Introduction – 5% of total marks** | | | |
| Excellent introduction to purposes of the document and the system to be developed and who it will serve. |  | Unsatisfactory / missing/ low value introduction. | |
| **Design Thinking – 10& of total marks** | | | |
| Excellent, accurate description of design thinking and how it applies here. Convincing evidence of use of additional resources (beyond what you have been given). Clear evidence of critical judgement. |  | Unsatisfactory coverage of design thinking and how it applies here. Little relevant material and/or Inaccurate answer or incomplete. disorganised, largely irrelevant material and misunderstanding,  minimal or no relevant material | |
| **Scenarios – 30% of total marks** | | | |
| Excellent, systematic, accurate description of relevant key scenarios as derived from design thinking. Convincing evidence of insight, critical evaluation and innovation.. |  | Unsatisfactory coverage of scenarios.  Inaccurate answer or incomplete. Disorganised, largely irrelevant scenarios and/or misunderstanding, | |
| **Logical view – 10% of total marks** | | | |
| . Excellent, systematic, description of planned system with clear notation. Convincing evidence of insight into software architecture and display of critical judgement. Clear evidence of independence of thought and originality. |  | Unsatisfactory coverage of planned system/ poor notation.  Answer incomplete. Disorganised, largely irrelevant /inappropriate architecture presented scenarios and/or misunderstanding, | |
| **Physical View – 5% of total marks** | | | |
| Excellent, systematic description of deployment of planned system with clear notation. Convincing evidence of insight into deployment options |  | Unsatisfactory description of deployment of planned system/poor notation.  Answer incomplete. Disorganised, largely irrelevant /inappropriate deployment and/or misunderstanding | |
| **Process View – 20% of total marks** | | | |
| Excellent, systematic, tailored description of quality attributes and tactics. Clear notation. Convincing evidence of insight and critical judgement. Clear evidence of independence of thought and originality |  | | Unsatisfactory coverage of quality attributes and tactics Answer incomplete. Disorganised, largely irrelevant /inappropriate and/or misunderstanding, |
| **Development View – 15% of total marks** | | | |
| Excellent systematic description of how the system will be implemented, the software process to be used, work division and project management protocols.  Clear convincing evidence of planning and team work. Innovative approach clearly demonstrated. Critical judgement demonstrated |  | | Unsatisfactory coverage of how the system will be implemented, the software process to be used, work division and project management protocols. Answer incomplete. Disorganised, largely irrelevant /inappropriate and/or misunderstanding, |